

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:)

Takeshi SHIMOYAMA)

Serial No.: To be assigned)

Filed: December 19, 2000)

Group Art Unit: Unassigned

Examiner: Unassigned

For: **A METHOD AND APPARATUS FOR DESIGNING CIPHER LOGIC, AND A
COMPUTER PRODUCT**

**SUBMISSION OF CERTIFIED COPY OF PRIOR FOREIGN
APPLICATION IN ACCORDANCE
WITH THE REQUIREMENTS OF 37 C.F.R. §1.55**

*Assistant Commissioner for Patents
Washington, D.C. 20231*

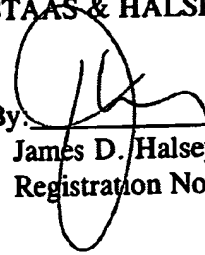
Sir:

In accordance with the provisions of 37 C.F.R. §1.55, the applicant submits herewith a certified copy of the following foreign application:

Japanese Patent Application No. 2000-016413
Filed: January 26, 2000

It is respectfully requested that the applicant be given the benefit of the foreign filing date as evidenced by the certified papers attached hereto, in accordance with the requirements of 35 U.S.C. §119.

Respectfully submitted,
STAAS & HALSEY LLP

By: 
James D. Halsey, Jr.
Registration No. 22,729

700 11th Street, N.W., Ste. 500
Washington, D.C. 20001
(202) 434-1500
Date: December 19, 2000

日 本 国 特 許 庁

PATENT OFFICE
JAPANESE GOVERNMENT

JC960 U.S. PTO
09/739219
12/19/00

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日
Date of Application:

2000年 1月26日

出 願 番 号
Application Number:

特願2000-016413

出 願 人
Applicant (s):

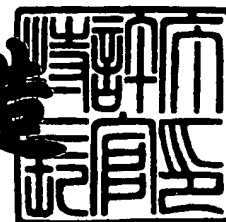
富士通株式会社

CERTIFIED COPY OF
PRIORITY DOCUMENT

2000年 9月18日

特許庁長官
Commissioner,
Patent Office

及 川 耕 造



出証番号 出証特2000-3073953

【書類名】 特許願

【整理番号】 0050031

【提出日】 平成12年 1月26日

【あて先】 特許庁長官 殿

【国際特許分類】 G06F 15/00

【発明の名称】 暗号設計装置および記録媒体

【請求項の数】 5

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

【氏名】 下山 武司

【特許出願人】

【識別番号】 000005223

【氏名又は名称】 富士通株式会社

【代理人】

【識別番号】 100089141

【住所又は居所】 東京都目黒区平町1丁目21番20-603号

【弁理士】

【氏名又は名称】 岡田 守弘

【電話番号】 03-3725-2215

【手数料の表示】

【予納台帳番号】 015543

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9705795

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 暗号設計装置および記録媒体

【特許請求の範囲】

【請求項 1】

計算機に最適な入出力数の $S - b \circ x$ を持つ暗号化装置を設計する暗号設計装置において、

計算機の一次キャッシュメモリ量、および全体の入出力ビット数を入力する手段と、

上記入力された全体の入出力ビット数を分割して $S - b \circ x$ の入出力数を生成し、余りがでたときに任意の $S - b \circ x$ の入出力数に割り当てて $S - b \circ x$ の入出力数を仮決定する手段と、

上記仮決定した $S - b \circ x$ の入出力数を組み合わせて、上記入力された一次キャッシュサイズを越えない範囲で当該組み合わせを行う手段とを備えたことを特徴とする暗号設計装置。

【請求項 2】

上記分割して算出した $S - b \circ x$ の入出力数の最小値を予め指定したことを特徴とする請求項 1 記載の暗号設計装置。

【請求項 3】

上記全体の入出力ビット数および上記一次キャッシュサイズできまる終了値をもとに、上記組み合わせを終了させることを特徴とする請求項 1 記載の暗号設計装置。

【請求項 4】

上記余りがでたときに可及的に離れた位置の $S - b \circ x$ の入出力数に割り当てることを特徴とする請求項 1 記載の暗号設計装置。

【請求項 5】

計算機の一次キャッシュメモリ量、および全体の入出力ビット数を入力する手段と、

上記入力された全体の入出力ビット数を分割して $S - b \circ x$ の入出力数を算出し、余りがでたときに任意の $S - b \circ x$ の入出力数に割り当てて $S - b \circ x$ の入

出力数を仮決定する手段と、

上記仮決定した $S - b o x$ の入出力数を組み合わせて、上記入力された一次キャッシュサイズを越えない範囲で当該組み合わせを行う手段として機能させるプログラムを記録したコンピュータ読取可能な記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、計算機に最適な入出力数の $S - b o x$ を持つ暗号化装置を設計する暗号設計装置および記録媒体に関するものである。

【0002】

【従来の技術】

高度情報化社会において高速かつ安全な通信を実現するためには共通鍵ブロック暗号が欠かせない道具であり、通常ユーザが意識することなく、日常的に用いられている。一口に共通鍵暗号と言っても、さまざまな共通鍵暗号アルゴリズムが提案されている。その共通鍵ブロック暗号の多くは、Feistel構造とよばれる単純な繰り返し構造であり、更にその内部構造であるF関数とよばれる部分では、通常 $S - b o x$ と言われる入力ビット数の少ない非線形関数を並べて用い、更にその出力を組み合わせたものを線形関数を用いて拡散させるという方法が最も主流として用いられている。このF関数の内部構造は、一般にSPN (Substitution Permutation Network) 構造と呼ばれている。暗号の安全性の核となるのは $S - b o x$ であるため、 $S - b o x$ の設計は簡単ではない。また、多くの種類の $S - b o x$ を用いる場合にはそれだけ多くの記憶容量が必要となる。そのため開発コスト、必要メモリ量を抑え、あるいは構造を見やすくするために、これまでの $S - b o x$ としては全く同じものを繰り返し用いるか、あるいは入力サイズ、および出力サイズがそれぞれ等しい $S - b o x$ を使い回して使うことが多かった。

【0003】

【発明が解決しようとする課題】

共通鍵暗号の入力ビット数は通常64ビット、あるいは128ビットであるた

め、重複なく均等なサイズのS-boxを用いる場合には4ビット入力、あるいは8ビット入力といった 2^n ビット入力のS-boxしか考えられなかった。

【0004】

S-boxは暗号装置において、最も頻繁に参照されるため、暗号化速度に最も影響を与えやすく、そのためにS-boxを表す表全体が、計算機の持つ最も高速に参照可能な記憶装置（通常は1次キャッシュメモリ）内に収まるように設計することが望まれる。一方、S-boxの入力ビット数に応じてそのテーブルサイズは指数関数的に大きくなるため現実的に用いられることができるテーブルの大きさには上限がある。もし、記憶装置の容量を越えたテーブルを参照しなければならない場合、アクセス速度は数倍以上遅くなってしまう。暗号設計においては、こうした理由から実装上不利な状況を避けるために、 2^n 入力のS-boxの選択肢は、現実的には4ビットかあるいは8ビット入力しかとりえなかった。

【0005】

一方、最近の計算機が持つメモリ容量も年々増加する傾向にある。これらの状況から16ビット入力のS-boxを利用するのは、まだ時期尚早であるものの、8ビット入力のS-boxでは計算機のメモリ資源を十分に生かしているとは言えないという問題もあった。

【0006】

即ち、小さな入力ビット数を持つS-boxでは、ほぼ全ての計算機に対して、高速アクセス可能なメモリ内に収めることが可能であるが、分割したS-boxの総数が増えることが避けられず、メモリ内に記憶されたテーブルを参照する回数が増え、実行速度が低下するという問題があった。

【0007】

また、大きな入力ビット数を持つS-boxを使えば、S-boxの総数を減らすことができ、テーブル参照の回数を減らすことができるが、テーブル全体の大きさが大きくなるために、記憶装置に保持できないか、あるいは参照速度が低速な記憶装置にしか保持することができない。従って、1回のテーブル参照にかかる時間そのものが低下し、結果として全体の実行速度が低下するという問題が

あった。

【0008】

本発明は、これらの問題を解決するため、計算機毎の一次キャッシュメモリ量および全体の入出力ビット数をもとにS-boxの入出力数を最適化し、計算機毎に最適な高速な暗号化／復号化を実現することを目的としている。

【0009】

【課題を解決するための手段】

図1を参照して課題を解決するための手段を説明する。

図1において、入力手段2は、パラメータ（計算機の一次キャッシュメモリ量、全体の入出力数、S-boxの最小の入出力数など）を入力するものである。

【0010】

S-boxの最適化手段3は、S-boxの入出力数の最適化を行うものである。

次に、動作を説明する。

【0011】

入力手段2が計算機の一次キャッシュメモリ量、および全体の入出力ビット数を入力し、S-boxの最適化手段3が入力された全体の入出力ビット数を分割してS-boxの入出力数を生成し、余りがでたときに任意のS-boxの入出力数に割り当ててS-boxの入出力数を仮決定し、仮決定したS-boxの入出力数を組み合わせて、入力された一次キャッシュサイズを越えない範囲で当該組み合わせを行うようにしている。

【0012】

この際、分割して算出したS-boxの入出力数の最小値を予め指定するようにしている。

また、全体の入出力ビット数および一次キャッシュサイズできまる終了値をもとに、組み合わせを終了させるようにしている。

【0013】

また、余りがでたときに可及的に離れた位置のS-boxの入出力数に割り当てるようにしている。

従って、計算機毎の一次キャッシュメモリ量および全体の入出力ビット数をもとに $S-b \circ x$ の入出力数を最適化することにより、計算機毎に最適な高速な暗号化／復号化を行う装置を実現することが可能となる。

【0014】

【発明の実施の形態】

次に、図1から図11を用いて本発明の実施の形態および動作を順次詳細に説明する。

【0015】

図1は、本発明のシステム構成図を示す。

図1において、処理装置1は、図示外の記録媒体から読み出したプログラムを主記憶にローディングして起動し、以下に説明する各種処理を行うものであって、入力手段2、 $S-b \circ x$ の最適化手段3、 $S-b \circ x$ の生成手段4、F関数の生成手段5などから構成されるものである。

【0016】

入力手段2は、パラメータ（計算機の一次キャッシュメモリ量、全体の入出力数、 $S-b \circ x$ の最小の入出力数など）などを入力するものである。

$S-b \circ x$ の最適化手段3は、 $S-b \circ x$ の入出力数の最適化を行うものである（図2、図3、図7などを用いて後述）。

【0017】

$S-b \circ x$ の生成手段4は、最適化された $S-b \circ x$ の入出力数に従った $S-b \circ x$ を生成するものである（図2、図4、図7から図9などを用いて後述する）。

【0018】

F関数の生成手段5は、F関数を生成するものである（図2、図5、図7、図11などを用いて後述する）。

入力ファイル6は、計算機の一次キャッシュメモリ量などを格納した計算機パラメータファイル、 $S-b \circ x$ に関する各種データを格納した $S-b \circ x$ のパラメータファイルなどである。

【0019】

出力ファイル7は、S - b o x（設計／実装）の出力ファイル、F関数（設計／実装）の出力ファイル、フェイステル構造データ（設計／実装）の出力ファイルなどである。

【0020】

表示装置8は、画面などを表示するものである。

入出力装置9は、各種入出力装置であって、プリンタ、ディスク装置などである。

【0021】

次に、図2のフローチャートの順番に従い図1の構成の動作を説明する。

図2は、本発明の全体動作説明フローチャートを示す。

図2において、S1は、計算機のパラメータの入力を行う。ここでは、計算機のパラメータとして図示の下記を入力する。

【0022】

・一次キャッシュメモリ量：

①P e n t i u m I I : 1 6 キロバイト

②P A - R I S C : 1 メガバイト

S2は、S - b o x および入出力全体のパラメータを入力する。例えば図示の下記を入力する。

【0023】

・S - b o x の入出力数：5ビット以上

・全体の入出力数：32ビット

S3は、S - b o x の最適化を行う。これは、第1に、S2で指定されたS - b o x の入出力数の最小値例えば5ビットで全体の入出力数32ビットを分割して6個の5ビット

5 5 5 5 5 5

と、余り2ビットとし、この余り2ビットを可及的に離れた位置、例えば左端と右端に割り当てて、

6 5 5 5 5 6

とする。

【0024】

第2に、2つづつを組にして

11 10 11

と3つに合成し、合成した後の組み合わせ数（ここでは3つ）が、後述する図3で説明する $a = ((\text{入出力ビット数}) / (\log_2 (\text{キャッシュサイズ})))$ の整数部分 $+ 1 = ((32) / (\log_2 (16000)))$ の整数部分 $+ 1 = 3$ となり、等しいので合成の最適化を終了する（S1の①の一次キャッシュメモリ量が16キロバイトの場合）。一方、S1の②の一次キャッシュメモリ量が1メガバイトの場合には、 $a = 2$ となり、S3の右側の③に示すように、3つづつを組にして合成し、

16 16

の2組とし、 $a = 2$ と等しいので最適化を終了する（S1の②の一次キャッシュメモリが1メガバイトの場合）。

【0025】

S4は、S-boxの生成を行う。これは、S3で最適化した後のS-boxの入出力数をもとにS-boxを生成する（図4を用いて後述する）。

S5は、F関数の生成を行う。これは、S4で生成したS-boxをもとに後述する図5で説明するようにしてF関数（F関数に関する秘密鍵、F関数入出力ビット数と等しいビット数を持つ鍵など）を作成する。

【0026】

S6は、暗号の生成を行う。

以上によって、計算機などのパラメータ（一次キャッシュメモリ量、全体の入出力ビット数）を入力すると、自動的にS-boxの入出力ビット数の最適化を行い、S-boxの生成、F関数の生成、および暗号の生成を行うことが可能となる。

【0027】

図3は、本発明の最適化手順フローチャートを示す。

図3において、S11は、5ビットを32を越えないだけ並べる。これは、既述した図2のS2で入力したパラメータ（S-boxの入出力数5ビット以上）

から最小値を5ビットとし、当該S-boxの入出力数の最小値5ビットで全体の入出力数32ビットを分割して当該32ビットを越えない分だけ並べる。ここでは、5ビットを6つ下記のように並べる（余りは2である）。

【0028】

5 5 5 5 5 5

S12は、余りを任意の場所に割り当てる。これは、S11で並べた余り（ここでは、2）を任意の場所、ここでは、できるだけ離れた場所である、左端と右端にそれぞれ1を割り当てて下記のようにする。

【0029】

6 5 5 5 5 6

S13は、キャッシュサイズを越えない範囲の組み合わせを求める。これは、S12で並べたS-boxの各入出力数を例えば左端から2つつつ組（あるいは3つつつ組）に、下記のようにする。

【0030】

11 10 11（2つつつ組の場合）

16 16（3つつつ組の場合）

S14は、 $a = ((\text{全体の入出力ビット数}) / (\log_2(\text{キャッシュサイズ})))$ の整数部分 + 1 を求める。例えば

$$\begin{aligned} a &= (32) / (\log_2(16000)) \text{ の整数部分} + 1 \\ &= (32 / 12) \text{ の整数部分} + 1 \\ &= (2.6 \text{ の整数部分}) + 1 \\ &= 3 \end{aligned}$$

として求める。

【0031】

S15は、S13で組み合わせた後の組み合わせ数bとS14で求めたa（組み合わせ数の終了値）とを比較する。b = aの場合（S13の組み合わせ数bが終了値a（例えば3）に等しくなった場合）にはS16に進み、最適化終了する。一方、b > aの場合には、S11に戻り組み合わせを繰り返す。

【0032】

以上によって、パラメータで指定された最小の $S-b \circ x$ の入出力数（例えば 5 ビット）で全体の入出力数（例えば 32 ビット）を分割して並べ、余りがでたときは可及的に離れた位置に割り当てて仮の $S-b \circ x$ の入出力数の組み合わせを作成し、組み合わせ数 b が全体の入出力ビット数およびキャッシュサイズから求めた終値 a に等しくなるまで組み合わせを繰り返し、最適化を行う。これにより、一次キャッシュメモリに収まる範囲内で $S-b \circ x$ の組み合わせ数を最小にして参照回数を削減し、計算機毎に個別に最適化を実現することが可能となる。

【0033】

図4は、本発明の $S-b \circ x$ の生成フローチャートを示す。

図4において、S21は、最適化後の割りあて数を抽出する。例えば図2のS3の①の最適化の場合には、6，5を抽出する。

【0034】

S22は、各々の割りあて数に対応する入出力ビット数を持つ非線形型テーブルを作成する。例えば右側に示すように、入力数5ビットをアドレスとし、出力が5ビットの非線形のテーブルを作成する。同様に、6ビットなどの非線形テーブルを作成する。

【0035】

以上によって、最適化後の $S-b \circ x$ の組み合わせ数に対応する非線形テーブルを作成できたこととなる。

図5は、本発明のF関数例（設計時）を示す。F関数として、図示のように、図4で作成した $S-b \circ x$ （非線形テーブル）を図中のSのように並べて接続し、上部の全体の入出力数（例えば32ビット）にXOR（排他論理和）回路を入れて鍵（例えば32ビット）との排他論理和演算を行った後の32ビットをそれぞれ各 $S-b \circ x$ に図示のように分割して接続する。各 $S-b \circ x$ からの出力ビットをまとめてここでは32ビットにし、更に、L（線形変換回路）を通して32ビットを出力する。

【0036】

以上の構成により、計算機毎に最適化した入出力数を持つ $S-b \circ x$ を用いたF関数を生成（設計）することが可能となる。

図6は、本発明のFeistel構造データ例を示す。これは、既述した図5で生成（設計）したF関数を用いて生成したFeistel構造データをイメージ的に示す。上部から平文（あるいは暗号文）が入力され、矢印のように処理結果が順に流れて下方から暗号文（あるいは平文）が出力されるものであって、暗号化あるいは復号化する回路として動作するものである。この際、図示の構造中のF関数を構成するS-boxの参照が、各計算機毎の一次キャッシュメモリ上でアクセスできるように最適化されているので、当該計算機毎に固有の一次キャッシュメモリ量を最大限有効に活用して高速に暗号化あるいは復号化を実行することが可能となる。

【0037】

次に、図7から図11を用いて計算機に実装時に動的に最適化を行うときの動作を順次詳細に説明する。

図7は、本発明の動作説明フローチャート（実装時）を示す。

【0038】

図7において、S21は、計算機のパラメータの入力を行う。これは、本願発明が実装された計算機のパラメータとして下記を入力する（読み込む）。

・一次キャッシュメモリ量：

PentiumII：16キロバイト

S22は、S-boxの最適組の抽出を行う。これは、既述した設計時と同様に、例えば初回は、S-boxの入出力数の最小値である例えば5ビットで全体の入出力数例えば32ビットを分割して並べ、余りがある場合には可及的に離れた位置に割り当てて

6 5 5 5 5 6

のS-boxの入出力数の組み合わせを作る。2回目以降は、既述した図3の最適化手順フローチャートに従い2つつつ、あるいは3つつつを組み合わせたりなどして最適な組み合わせを作り、最適化されるまで繰り返す。

【0039】

S33は、S-boxの合成テーブルを作成する。これは、右側に示すように、S32で抽出したS-boxの入出力数の合成テーブル、例えば初回は6，5

の S-box の合成テーブルをそれぞれ作成し、2 回目な 6 と 5 を組にした 11 ビットの合成テーブル（拡大 S-box）などをそれぞれ作成することを繰り返す。

【0040】

S34 は、終わりか判別する。YES の場合には、S35 に進む。NO の場合には、S32 に戻り、次の最適な組の抽出を行うなどを繰り返す。

S35 は、各々拡大された S-box と L（線形変換回路）とを合成する。これにより、L（線形変換回路）が各々の拡大 S-box に取り込まれ、当該 L（線形変換回路）の処理が不要となり、高速化を図ることが可能となる。

【0041】

S36 は、残りの部分を実装する。例えば右側に記載したように、鍵加算部、入出力部などの残りの部分を実装する。

S37 は、F 関数の実装が終了したこととなる。

【0042】

以上によって、計算機への実装時に当該計算機からパラメータ（一次キャッシュメモリ量など）を取り込み、この一次キャッシュメモリ量と全体の入出力数をもとに最適な S-box の入出力数を求めて F 関数を自動生成することが可能となる。これにより、当該 F 関数を組み込んだ既述した図 6 の Feistel 構造データを作成し、平文／暗号文を暗号化／復号化するときに当該計算機が持つ一次キャッシュメモリ上に配置したテーブルを必要最小限の参照回数にして高速化を図ることが可能となる。

【0043】

図 8 は、本発明の S-box の抽出フローチャートを示す。

図 8 において、S41 は、最適化された S-box の組を抽出する。例えば右側に示す下記のように最適化された S-box の入出力数の組を抽出する。

【0044】

6 5 5 5 5 6

S42 は、キャッシュメモリ量に応じて組み合わせを抽出する。これは、S41 で抽出した S-box の入出力数の組を例えば左端から 2 つずつ、あるいは 3

つづつ組み合わせて新たな組を作成し、既述した図 3 の S 1 4 の a の終値と等しい組み合わせ数となるまで繰り返し、最適な組み合わせ数を持つ S - b o x の入出力数を決定する。例えば図示の下記のように決定（抽出）する。

【 0 0 4 5 】

1 1 1 0 1 1

以上によって、計算機への実装時に当該計算機の一次キャッシュメモリ量に対応した最適な S - b o x の入出力数を決定することが可能となる。

【 0 0 4 6 】

図 9 は、本発明の合成テーブルの作成フローチャートを示す。

図 9 において、S 5 1 は、S - b o x の組み合わせを入力する。これは、右側に記載したように、例えば S - b o x の入出力数として 6, 5 を入力する。

【 0 0 4 7 】

S 5 2 は、S - b o x の合成を行う。これは、S 5 1 で入力された S - b o x の合成を行い、右側に記載したように、既述した拡大 S - b o x を作成する。ここでは、例えば右側に示すように、入力数 1 1 ビットをアドレスとし、出力が 1 1 ビットの非線形のテーブルを作成する。

【 0 0 4 8 】

以上によって、最適化後の S - b o x の組み合わせ数に対応する合成テーブル（非線形テーブル）を作成できたこととなる。

図 1 0 は、本発明の S - b o x と L の合成フローチャートを示す。

【 0 0 4 9 】

図 1 0 において、S 6 1 は、拡大された S - b o x を入力する。

S 6 2 は、線形変換 L を入力する。

S 6 3 は、拡大 S - b o x の出力の、L による線形変換の結果を格納する。これは、例えば右側に示すように、拡大 S - b o x の出力を、線形変換 L で変換した後の結果を出力するように、右側のテーブルに格納し、当該線形変換 L の処理を拡大 S - b o x のテーブルに取り込む。

【 0 0 5 0 】

以上によって、線形変換 L の処理が拡大 S - b o x に取り込まれ、暗号化／復

号化時に当該線形変換Lの処理が不要となり、高速化を図ることが可能となる。

図11は、本発明のF関数例（実装時）を示す。ここでは、図10で合成した後の拡大S-boxである S_{11} 、 S_{10} 、 S_{11} を図示のように配置および他の回路（XORなど）を配置することにより、F関数を生成することが可能となる。そして、当該生成したF関数をもとに既述した図6のFeistel構造データを作成し、平文を暗号化して暗号文を出力したり、暗号文を復号化して平文を出力したりすることが可能となる。この際、計算機が持つ一次キャッシュメモリ上にテーブルを配置して参照回数を最小限にして高速に暗号化／復号化の処理を実行することが可能となる。

【0051】

【発明の効果】

以上説明したように、本発明によれば、計算機毎の一次キャッシュメモリ量および全体の入出力ビット数をもとにS-boxの入出力数を最適化する構成を採用しているため、計算機毎に最適な高速な暗号化／復号化を行う装置を設計したり、実装したりすることが可能となる。

【図面の簡単な説明】

【図1】

本発明のシステム構成図である。

【図2】

本発明の全体動作説明フローチャートである。

【図3】

本発明の最適化手順フローチャートである。

【図4】

本発明のS-boxの生成フローチャートである。

【図5】

本発明のF関数例（設計時）である。

【図6】

本発明のFeistel構造データ例である。

【図7】

本発明の動作説明フローチャート（実装時）である。

【図 8】

本発明の S - b o x の抽出フローチャートである。

【図 9】

本発明の合成テーブルの作成フローチャートである。

【図 1 0】

本発明の S - b o x と L の合成フローチャートである。

【図 1 1】

本発明の F 関数例（実装時）である。

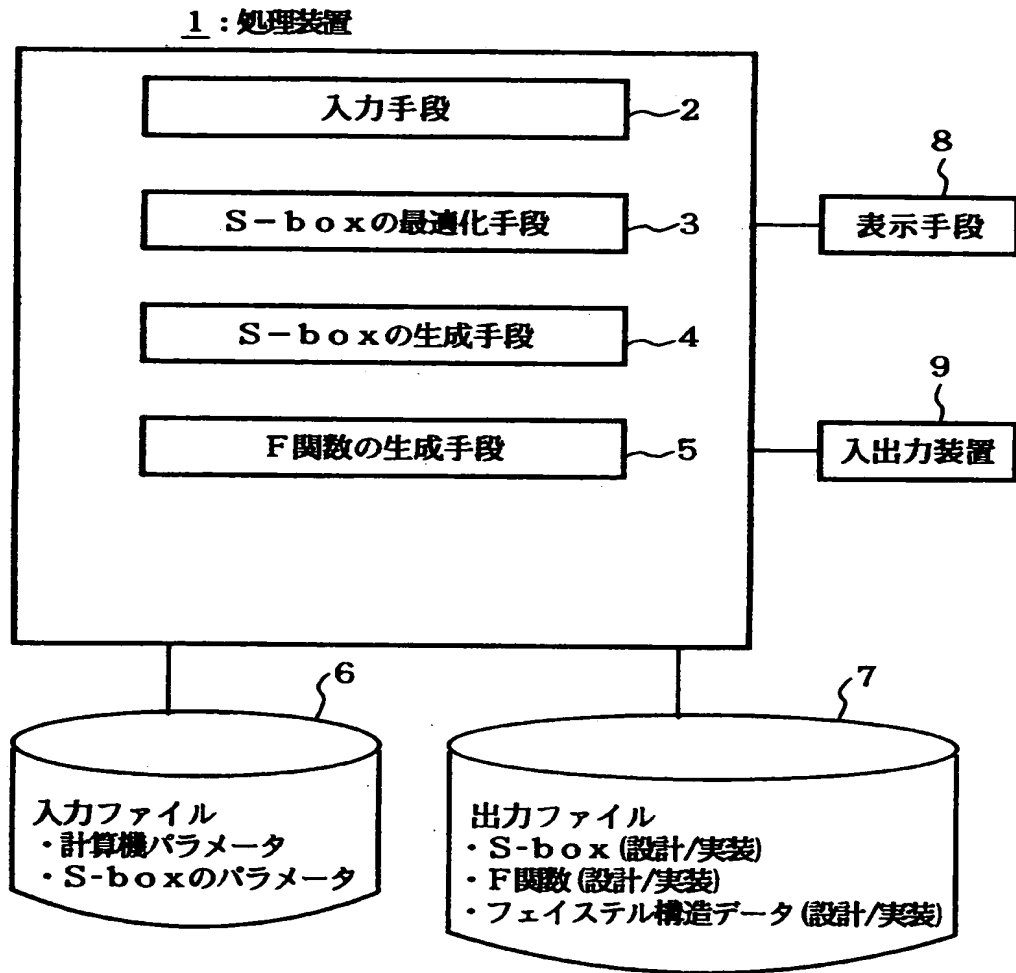
【符号の説明】

- 1 : 処理装置
- 2 : 入力手段
- 3 : S - b o x の最適化手段
- 4 : S - b o x の生成手段
- 5 : F 関数の生成手段
- 6 : 入力ファイル
- 7 : 出力ファイル
- 8 : 表示装置
- 9 : 入出力装置

【書類名】 図面

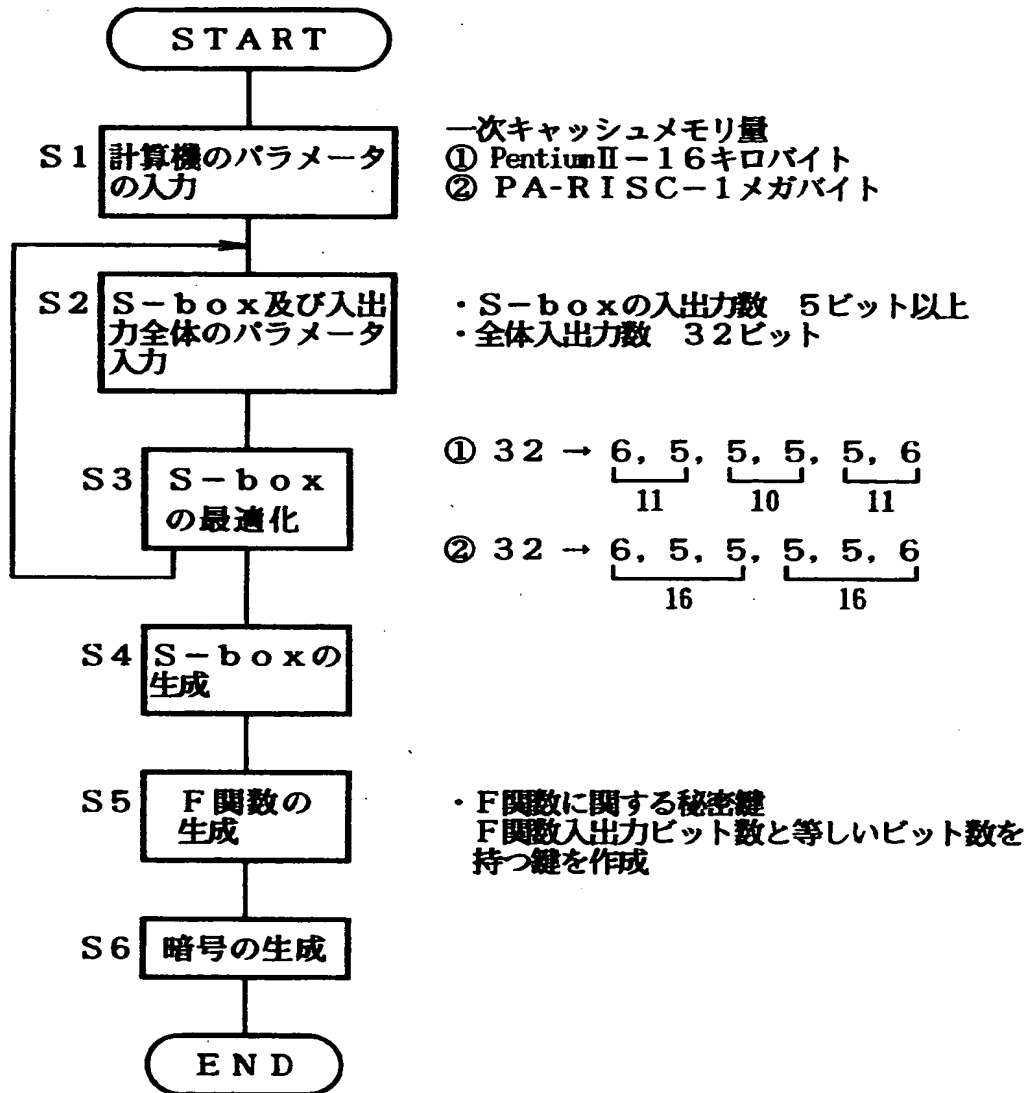
【図 1】

本発明のシステム構成図



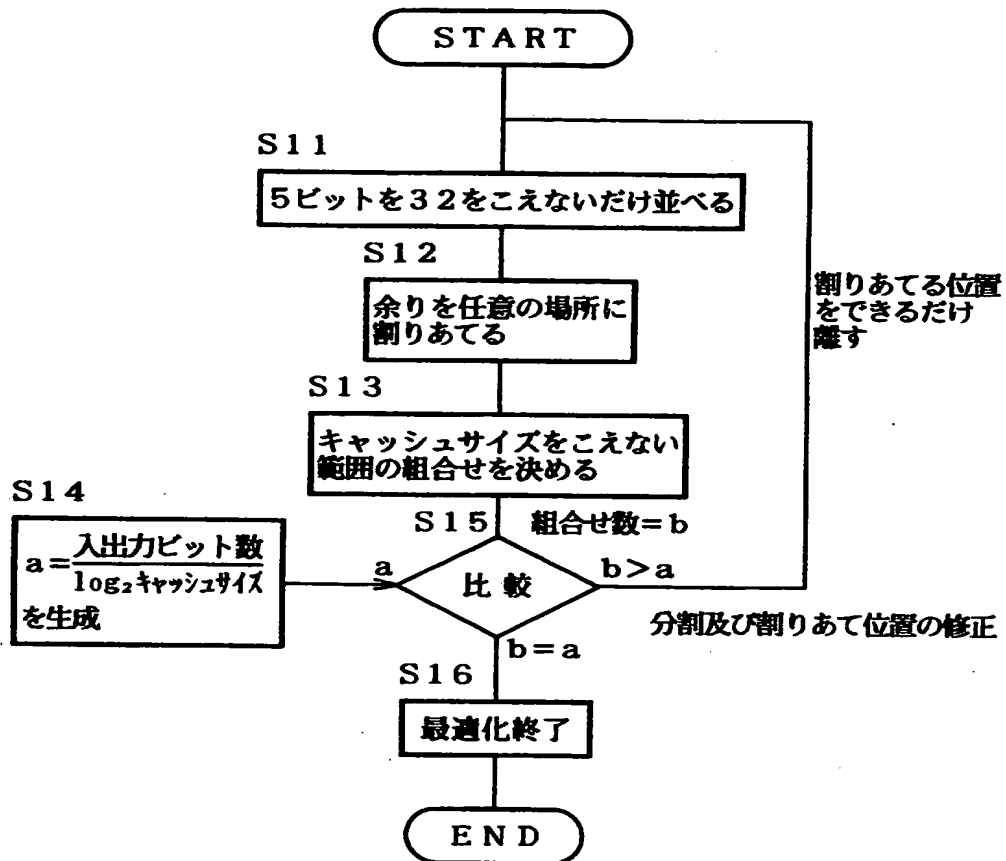
【図 2】

本発明の全体動作説明フローチャート

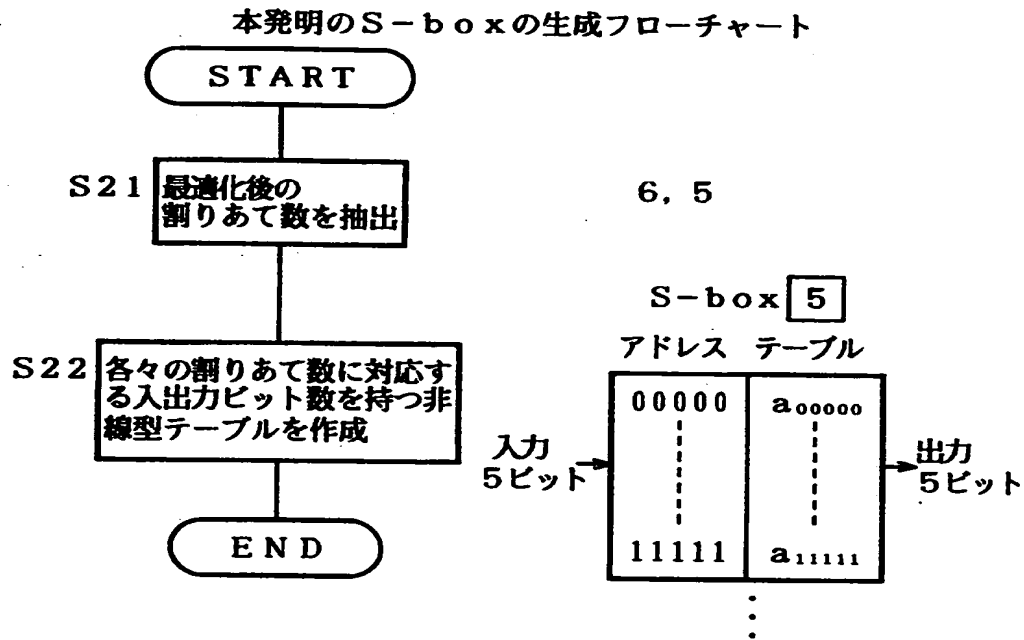


【図 3】

本発明の最適化手順フローチャート

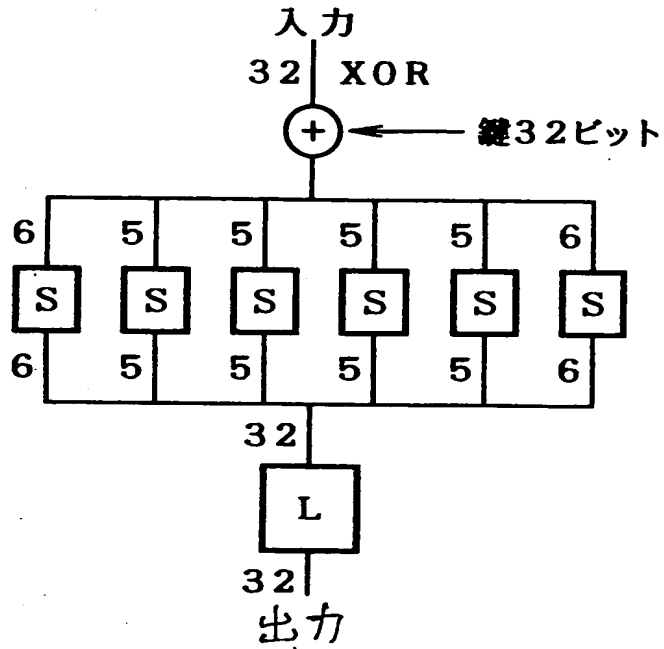


【図 4】



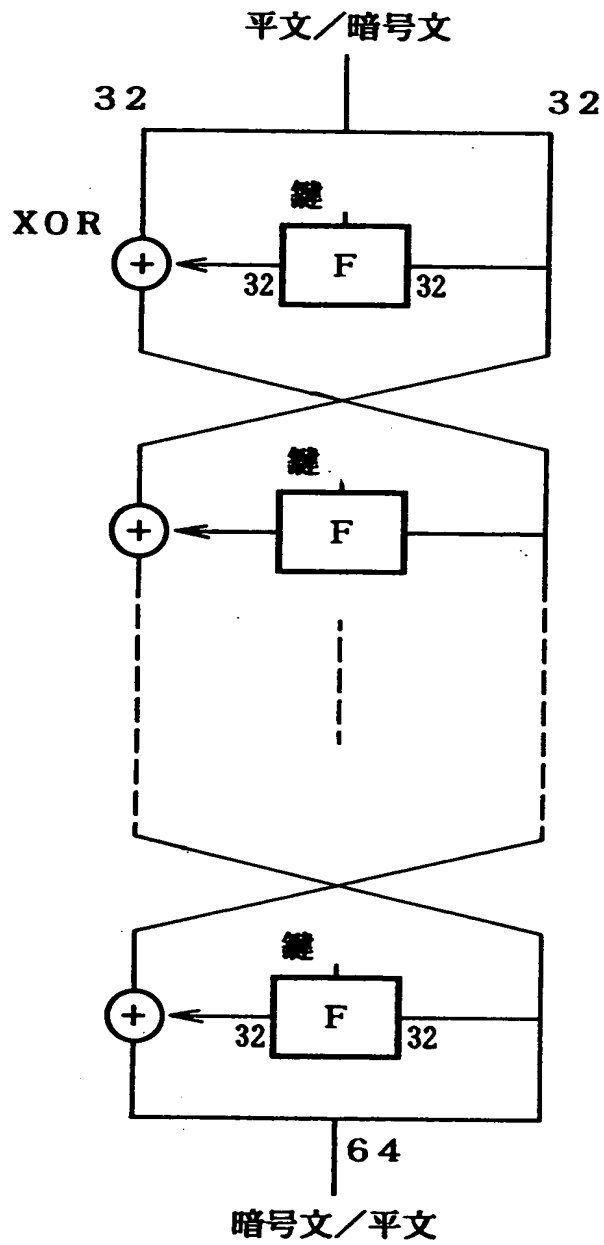
【図 5】

本発明の F 関数例（設計時）



【図6】

本発明のFeistel構造データ例



START

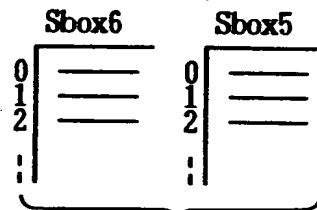
S31 計算機の パラメータ入力

一次キャッシュメモリ量
Pentium II-16キロバイト

S32 S-boxの最適組の抽出

655556
(6, 5)

S33 S-boxの合成テーブル作成



S34

終りか？

Sbox11

0	—
1	—
2	—
⋮	

拡大Sbox

S35 各々拡大された
S-boxとLとを合成

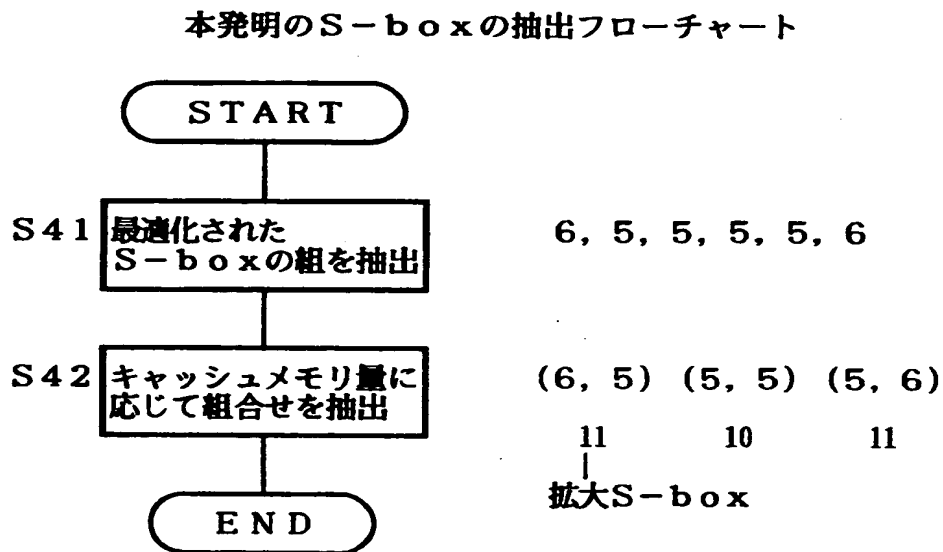
**S36 残りの部分を
実装**

- ・鍵加算部
- ・入出力部

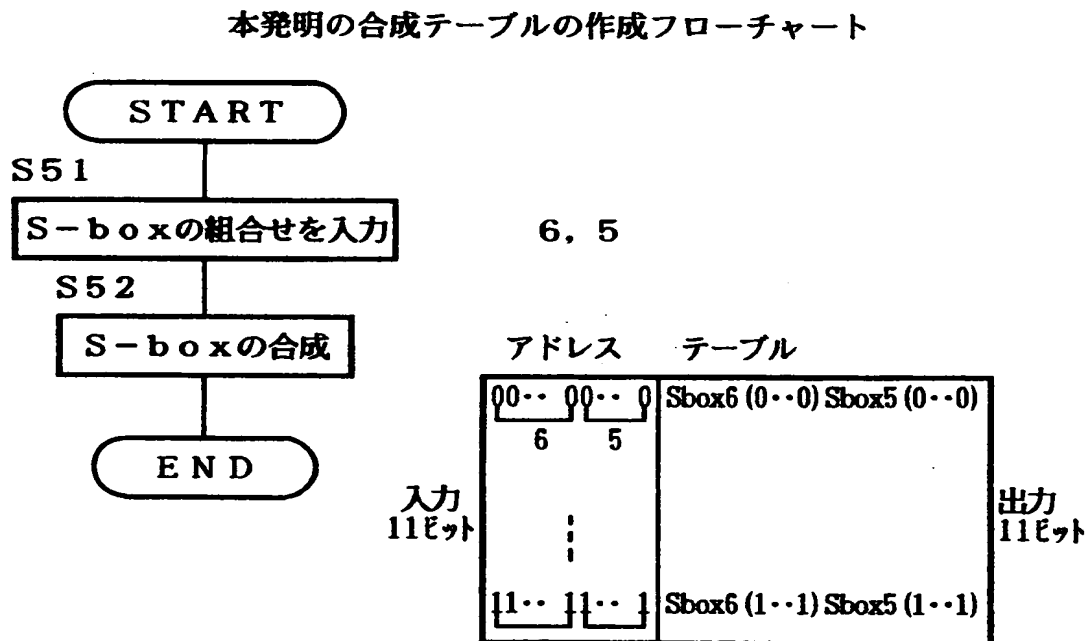
S37 F関数の
実装終了

END

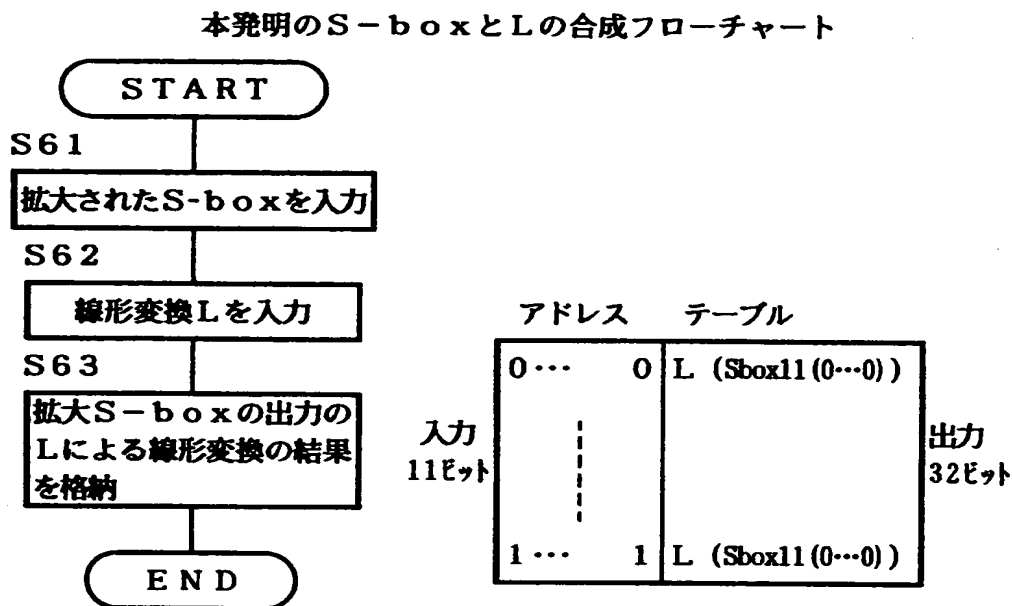
【図 8】



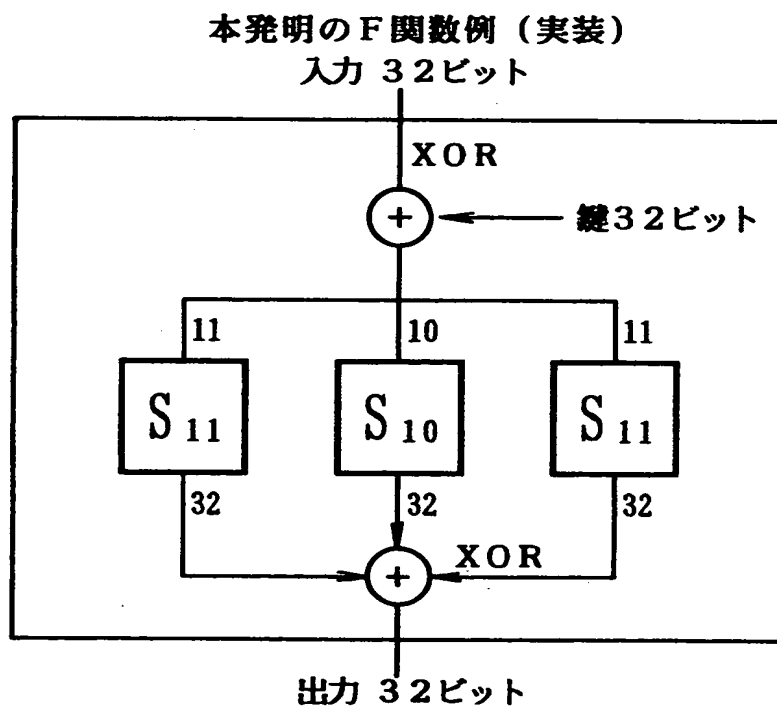
【図 9】



【図 1 0】



【図 1 1】



【書類名】 要約書

【要約】

【課題】 本発明は、計算機に最適な入出力数の $S - box$ を持つ暗号化装置を設計する暗号設計装置および記録媒体に関し、計算機毎の一次キャッシュメモリ量および全体の入出力ビット数をもとに $S - box$ の入出力数を最適化し、計算機毎に最適な高速な暗号化／復号化を実現することを目的とする。

【解決手段】 計算機の一次キャッシュメモリ量、および全体の入出力ビット数を入力する手段と、入力された全体の入出力ビット数を分割して $S - box$ の入出力数を生成し、余りがでたときに任意の $S - box$ の入出力数に割り当てて $S - box$ の入出力数を仮決定する手段と、仮決定した $S - box$ の入出力数を組み合わせて、入力された一次キャッシュサイズを越えない範囲で当該組み合わせを行う手段とから構成する。

【選択図】 図 1

出 願 人 履 歴 情 報

識別番号 [000005223]

1. 変更年月日	1996年 3月26日
[変更理由]	住所変更
住 所	神奈川県川崎市中原区上小田中4丁目1番1号
氏 名	富士通株式会社